

CLAIMS

1 1. A method for executing a load locked and a store conditional instruction in
2 a processor, comprising:

3
4 executing the load locked instruction to read a memory block, and said processor
5 in response to executing the load locked instruction issuing a read modify system com-
6 mand to read said block and to take ownership of said block by said processor, and also
7 setting a lock flag for the address of said memory block, and writing a value of said
8 memory block into a cache of said processor as a cache copy of said memory block;

9
10 resetting said lock flag upon receipt of an invalidate message for said cache copy
11 of said memory block, if any said invalidate messages are received by said processor;

12
13 waiting for a selected time interval before said processor surrenders ownership of
14 said memory block upon receipt of an ownership request message, if any is received by
15 said processor after execution of said load locked instruction;

16
17 executing the store conditional instruction, and said processor in response to exe-
18 cuting said store conditional instruction testing said lock flag, and if said lock flag is set,
19 writing to said cache copy of said memory block.

1 2. The method as in claim 1 further comprising:

2
3 checking the cache before executing the load locked instruction, and in the event
4 that there is a cache miss, then proceeding to execute said load locked instruction, and in
5 the event that there is a cache hit then reading the value from its cache and setting said
6 lock flag.

1 3. The method as in claim 1 further comprising:

2 said invalidate messages are Inval Probes.

1 4. The method as in claim 1 further comprising:
2 said invalidate messages are forwarded RdMod probes.

1 5. The method as in claim 1 wherein said executing said store conditional instruc-
2 tion further comprises:
3 ending, in the event that said lock flag is reset, said store conditional instruction
4 and not writing to said cache copy of said memory block.

1 6. The method as in claim 1 wherein said selected time interval further comprises:
2 a first time required for a predetermined number of cycles of a clock in said proc-
3 essor to retire.

1 7. The method as in claim 1 wherein said selected time interval further com-
2 prises:
3 a second time required for a predetermined number of assembly language instructions
4 executing in said processor to retire.

1 8. The method as in claim 1 wherein said selected time interval further com-
2 prises:
3 a third time required until a store conditional instruction referring to said memory
4 block executes.

1 9. The method as in claim 1 wherein said selected time interval further com-
2 prises:
3 a fourth time for all MAF entries in said processor MAF file to retire.

1 10. The method as in claim 1 wherein said selected time interval further com-
2 prises:
3 a fifth time during which a MAF entry for the LD_L instruction persists in a MAF
4 file of said processor.

1 11. A computer executing a load locked and store conditional instructions, com-
2 prising:

3
4 a processor executing the load locked instruction to read a memory block, and
5 said processor in response to executing the load locked instruction issuing a read modify
6 system command to read said block and to take ownership of said block by said proces-
7 sor, and also setting a lock flag for the address of said memory block, and writing a value
8 of said memory block into a cache of said processor as a cache copy of said memory
9 block;

10
11 a MAF state machine resetting said lock flag upon receipt of an invalidate mes-
12 sage for said cache copy of said memory block, if any said invalidate messages are re-
13 ceived by said processor;

14
15 a controller waiting for a selected time interval before said processor surrenders
16 ownership of said memory block upon receipt of an ownership request message, if any is
17 received by said processor after execution of said load locked instruction;

18
19 said processor executing the store conditional instruction, and said processor
20 testing said lock flag, and if said lock flag is set, writing to said cache copy of said mem-
21 ory block.

1 12. The apparatus as in claim 11 further comprising:

2
3 said processor checking the cache before executing the load locked instruction,
4 and in the event that there is a cache miss then proceeding to execute said load locked
5 instruction, and in the event that there is a cache hit then reading the value from its cache
6 and setting said lock flag.

1 13. The apparatus as in claim 11 further comprising: said invalidate message is
2 an Inval Probe.

1 14. The apparatus as in claim 11 further comprising: said invalidate message is a
2 RdMod probe.

1 15. The apparatus as in claim 11 further comprising:
2 said processor ending, in the event that said lock flag is reset, said store condi-
3 tional instruction and not writing to said cache copy of said memory block.

1 16. The apparatus as in claim 11 further comprising:
2 a first time required for a predetermined number of cycles of a clock in said proc-
3 essor to retire.

1 17. The apparatus as in claim 11 further comprising:
2 a second time required for a predetermined number of assembly language instruc-
3 tions executing in said processor to retire.

1 18. The apparatus as in claim 11 further comprising:
2 a third time required until a store conditional instruction referring to said memory
3 block executes.

1 19. The apparatus as in claim 11 further comprising:
2 a fourth time for all MAF entries in said processor MAF file to retire.

1 20. The apparatus as in claim 11 further comprising:
2 a fifth time during which a MAF entry for the LD_L instruction persists in a MAF
3 file of said processor.

1 21. A method of operating a multiprocessor computer system, comprising:
2

3 executing a load locked and a store conditional instruction in each processor in
4 accordance with the following steps;

5 executing said load locked instruction to read a memory block, and said processor
6 in response to executing the load locked instruction issuing a read modify system com-
7 mand to read said block and to take ownership of said block by said processor, and also
8 setting a lock flag for the address of said memory block, and writing a value of said
9 memory block into a cache of said processor as a cache copy of said memory block;

11 resetting said lock flag upon receipt of an invalidate message for said cache copy
12 of said memory block, if any said invalidate messages are received by said processor;

14 waiting for a selected time interval before said processor surrenders ownership of
15 said memory block upon receipt of an ownership request message, if any is received by
16 said processor after execution of said load locked instruction;

18 executing the store conditional instruction, and said processor in response to exe-
19 cuting said store conditional instruction testing said lock flag, and if said lock flag is set,
20 writing to said cache copy of said memory block.

1 22. A multiprocessor computer system comprising:

3 a plurality of processor systems, each processor system having a processor exe-
4 cuting a load locked and store conditional instructions,

6 said processor executing the load locked instruction to read a memory block, and
7 said processor in response to executing the load locked instruction issuing a read modify
8 system command to read said block and to take ownership of said block by said proces-
9 sor, and also setting a lock flag for the address of said memory block, and writing a value
10 of said memory block into a cache of said processor as a cache copy of said memory
11 block;

13 a MAF state machine resetting said lock flag upon receipt of an invalidate mes-
14 sage for said cache copy of said memory block, if any said invalidate messages are re-
15 ceived by said processor;

16

17 a controller waiting for a selected time interval before said processor surrenders
18 ownership of said memory block upon receipt of an ownership request message, if any is
19 received by said processor after execution of said load locked instruction;

20

21 said processor executing the store conditional instruction, and said processor
22 testing said lock flag, and if said lock flag is set, writing to said cache copy of said mem-
23 ory block.

1 23. A computer readable media comprising: computer readable instructions for the
2 practice of the method of claim 1 written thereon.

1 24. Electromagnetic signals propagating on a computer network, said electromagnetic
2 signals carrying information for the practice of the method of claim 1.